

## 仮想メモリ・システムの二次記憶管理方式の比較解析†

西 垣 通<sup>††</sup> 緒 方 慎 八<sup>†††</sup>

従来、仮想メモリ・システムの性能は主に主記憶管理について論じられ、二次記憶管理に関する報告は知られていない。仮想空間上のページと二次記憶上のスロットとを対応づけるスロット割り当て方式は、ページ入出力実行時間に影響し、しばしばシステム全体の性能を左右する。広く用いられる方式として、各ページに終始同一のスロットを割り当てる固定方式と、ページ書き出し時点で動的にスロットを割り当てる浮動方式とがある。

本論文は、両方式の効率の理論的比較に関する。評価関数として、会話型ジョブの1インタラクションあたりのページ入出力実行時間を用いた。浮動方式のもとでは、ひとつのプログラムに割り当てられるスロット群は、そのプログラム特性にしたがって二次記憶上に散在する。この散在度を算定するためのモデルを導いた。具体的数値例により、散在度が小のとき浮動方式、大のとき固定方式が、各々優れることを示した。

### 1. ま え が き

仮想メモリ方式の計算機システムにおいては、仮想空間上の各ページに対し、二次記憶上のスロットとよばれるレコードが割り当てられる。この割り当て方式がページ入出力効率を左右し、システム全体の性能に多大な影響を与える。特に、コマンド到着時点でのswap-inの効率は、会話端末の応答性を決定する主要因のひとつとなる。しかし従来、仮想メモリ方式の性能解析は主として主記憶管理に関するものであり<sup>3)-5)</sup>、二次記憶管理に関する報告は知られていない。

スロット割り当て方式として従来より用いられるものに、ジョブの実行を通じて各ページに常に一定のスロットを対応させる固定方式<sup>9),10)</sup>と、ページ書き出し時点でその書き出し時間が最小となる位置のスロットを割り当てる浮動方式<sup>9),10)</sup>の2つがある。

本論文では、上記2方式のページ入出力効率を比較するための解析モデルを提示する。さらに具体的数値例にもとづいて両者の得失を比較する。一般のファイル入出力解析<sup>1),2)</sup>と異なる点は、一連の入出力要求が互いに独立ではなく、プログラム動作に依存して関連することである。プログラム動作モデルはVSLM(The Very Simple Locality Model)<sup>3)</sup>を採用し、主記憶管理方式はworking-set scheduling<sup>4),5)</sup>方式を仮定する。またアプリケーションは会話型ジョブを想定す

る。さらに二次記憶媒体としては、スロット位置の入出力効率への影響が比較的大なることから、ディスクを仮定する。

### 2. 二次記憶のスロット割り当て方式

本論文で用いる用語を整理する。「スロット」とは、二次記憶媒体であるディスク上の、ページ単位に区分されたレコードである。「swapping」とは、ジョブのworking-set<sup>5)</sup>を、二次記憶と主記憶の間でやりとりする行為\*である。「interaction」とは、インタラクティブな会話型ジョブの、端末入力終了時から次に端末入力待ちになるまでの処理を意味する。

本論文で性能を比較する2つのスロット割り当て方式は次の通りである。

- 固定方式<sup>9),10)</sup>: 各ページに対し、ジョブ実行開始から終了まで常に同一のスロットを割り当てる。ジョブ実行中に内容が変更されたページを書き出すときは、以前当該ページに割り当てられていたスロットに書き出す。1つのプログラムのページ群に対する割り当ては、1ないし隣接したシリンダ上になされる。

- 浮動方式<sup>9),10)</sup>: 各ページに対し、ジョブ実行中にスロットを動的に割り当てる。ジョブ実行中に内容が変更されたページを書き出すときは、書き出し時間が最小となるよう、現在ヘッドが停止しているシリンダないしその近傍シリンダの新たなスロットが割り当てられる。

上記両方式を比較するための評価関数を設定する。swappingの効率が性能上最も重要だが、これ以外のページ入出力として、page-fault(working-setへ加わ

† A Comparative Analysis of Secondary Storage Management Techniques in a Virtual Memory System by TOHRU NISHIGAKI (Systems Development Laboratory, Hitachi, Ltd.) and SHINPACHI OGATA (Software Works, Hitachi, Ltd.).

†† (株)日立製作所システム開発研究所

††† (株)日立製作所ソフトウェア工場

\* したがってページでなくプロセスの入れ替えを意味する。

るページの読みこみ)と page-out (working-set から外れ, かつ書きこみの行われたページの書き出し)の2つがある。この中で, 前者の読みこみ時間は, スロット割り当て方式による影響が小さい。多重プログラミングのもとでは, あるジョブが page-fault を発生したとき, その要求ページに対応するスロットの位置ならびにヘッドの位置は, ともにランダムと考えられるためである。したがって次式を評価関数とする。

$$C \triangleq C_{SO} + C_{SI} + C_{PO}. \quad (1)$$

$C_{SO}, C_{SI}, C_{PO}$  はそれぞれ swap-out cost, swap-in cost, page-out cost であり, 各々の場合のアクセス時間\*を表わす。C を total cost とよぶ。C の減少は, 二次記憶媒体の負荷減少であるから, ジョブの応答性のみならずシステム全体の処理能力の向上につながる。式(1)の右辺をさらに次のようにかける。

$$\begin{cases} C_{SO} \triangleq f_{SO} + g_{SO}. \\ C_{SI} \triangleq f_{SI} + g_{SI}. \\ C_{PO} \triangleq f_{PO} + g_{PO}. \end{cases} \quad (2)$$

$f, g$  は各々シーク時間, サーチ時間の期待値を表わす。

### 3. 固定方式の性能解析

固定方式における total cost  $C^{FX}$  を求めるための諸元を導く(以下,  $C, f, g$  などの右肩の FX は固定方式, FL は浮動方式を表わす)。

$$(1) \text{ swap-out cost } C_{SO}^{FX}$$

ディスクのシーク時間を図1のような一次関数で近似する。二次記憶がディスク上でLシリンダを占めるとき,  $f_{SO}^{FX}$  は次式で与えられる。

$$f_{SO}^{FX} \cong S_0 + KL/3. \quad (3)$$

ただしここで, ジョブに属するすべてのページは1シ

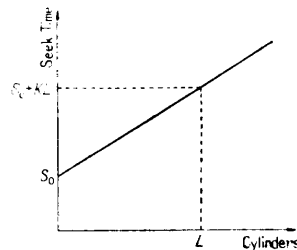


図1 ディスク・シーク時間の一次関数による近似  
Fig. 1 A continuous approximation for disk seek time.

\* ディスク占有時間はアクセス時間と転送時間の和だが, 転送時間は方式によらず一定である。

\*\* シリンダ内でのソートは行わないと仮定する。

リンダに収納されると仮定し, そのシリンダはLシリンダからランダムに選ばれるとする。swap-out 発生時のヘッド位置はランダムと考えられるから式(3)が得られる<sup>1)</sup>。また,  $g_{SO}^{FX}$  は次式で与えられる。

$$g_{SO}^{FX} \cong wp \cdot R/2. \quad (4)$$

$w$  は平均 working-set size である。 $p$  は, 1 interaction の間に working-set 内のあるページがチェンジする(書きこみで内容が変わる)確率である。 $R$  はディスクの回転時間であり,  $R/2$  はランダム・サーチ時間\*\*を表わす。

$$(2) \text{ swap-in cost } C_{SI}^{FX}$$

swap-out の場合と同様にして, シーク時間  $f_{SI}^{FX}$ , サーチ時間  $g_{SI}^{FX}$  は, 各々式(5), (6)で与えられる。

$$f_{SI}^{FX} \cong S_0 + KL/3. \quad (5)$$

$$g_{SI}^{FX} \cong w \cdot R/2. \quad (6)$$

$$(3) \text{ page-out cost } C_{PO}^{FX}$$

page-out は, ページが working-set から外れた時点で行うのが本来だが, 実際には効率を上げるために, 外れたページ群のうちチェンジしたものをある程度まとめて間歇的に書き出す。1 interaction の処理ステップ数は比較的小さいので(数十~数百kステップ), 1 interaction について page-out を一括して行うと仮定することができる。このとき,  $f_{PO}^{FX}, g_{PO}^{FX}$  は各々式(7), (8)で与えられる。

$$f_{PO}^{FX} \cong S_0 + KL/3. \quad (7)$$

$$g_{PO}^{FX} \cong wpr \cdot R/2. \quad (8)$$

ただし式(8)において,  $r$  は 1 interaction, 1 ページあたりの page-fault 発生率である(式(17)参照)。

### 4. 浮動方式の性能解析

#### 4.1 total cost $C^{FL}$

浮動方式における total cost  $C^{FL}$  を求めるための諸元を以下に導く。

$$(1) \text{ swap-out cost } C_{SO}^{FL}$$

swap-out 時のシーク時間  $f_{SO}^{FL}$ , サーチ時間  $g_{SO}^{FL}$  は, 各々式(9), (10)で与えられる。

$$f_{SO}^{FL} \cong 0. \quad (9)$$

$$g_{SO}^{FL} \cong R/2. \quad (10)$$

ただしここで, swap-out 時点でヘッドが停止しているシリンダ内に連続した  $wp$  個の未使用スロット群が常に存在すると仮定した。

(2) swap-in cost  $C_{SI}^{FL}$

$f_{SI}^{FL}, g_{SI}^{FL}$  は各々式(11), (12)で与えられる.

$$f_{SI}^{FL} \cong \{S_0 + KL/2\} + \{S_0 + KL/(y+1)\}y. \quad (11)$$

$$g_{SI}^{FL} \cong x \cdot R/2. \quad (12)$$

swap-out, page-out の際、ページはその時点でヘッドが停止していたシリンダ内にまとめて書き出される。このときの連続したスロット群を「ブロック」とよぶことにする。interaction をくり返す間に、あるプログラムのページ群に対応するスロット群はディスク上に散在する。xは、定常状態で、working-set 内のページ群に割り当てられたスロット群の属する相異なるブロック数の期待値である。また、yは、相異なるシリンダ数の期待値である。多重プログラミングのもとでは、x個のブロックはLシリンダ上にランダムに散在すると考えられるので、yはxの関数として次のように求まる(証明は文献6)のIV章 pp. 127-137を参照されたい)。

$$\left\{ \begin{aligned} y &= \sum_{m=\max(0, L-x)}^{L-1} (L-m) \cdot T_m(x, L) & (13) \\ \text{ただし } T_m(x, L) &= \binom{L}{m} \sum_{j=0}^{L-m} (-1)^j \binom{L-m}{j} \\ &\quad \times \left(1 - \frac{m+j}{L}\right)^x & (14) \end{aligned} \right.$$

ここで、 $T_m(x, L)$  は、ちょうど  $(L-m)$  個のシリンダにx個のブロックが存在し、m個のシリンダが空である確率である\*。次節でxをw, p, rから求める議論を述べるが、ここではxやyが与えられたとする。いま、アクセスすべきy個のシリンダがディスク上に散在しているとき、読み込むスロット群をシリンダ番号についてソートし、図2のようにスキャンすると仮

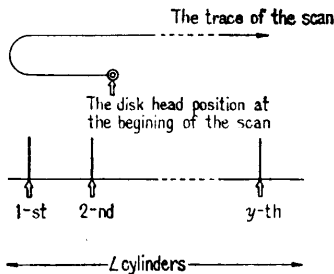


図2 浮動方式におけるスワップ・イン時の、y個のシリンダのアクセス方法  
Fig. 2 The scan of y cylinders to swap-in the working-set under Floating Technique.

\*  $T_m(x, L)$  は実際には漸化式により簡単に求まる。6)参照。  
\*\* 参照される確率の高いページの集合。  
\*\*\* interaction の途中で swap-out されることはないとする。

定する。式(11)の第1項はLシリンダの左端までのシーク時間期待値であり、第2項はy個のシリンダをスキャンするシーク時間の期待値<sup>1)</sup>である。また、サーチすべきブロック数はxであるから、式(12)が得られる。

(3) page-out cost  $C_{PO}^{FL}$

1 interaction ごとに一括して行われる page-out については、swap-out と同様にして次式が得られる。

$$f_{PO}^{FL} \cong 0. \quad (15)$$

$$g_{PO}^{FL} \cong R/2. \quad (16)$$

4.2 ブロック数xの算定

xの算定にあたり、プログラムの動作はVSLM<sup>3)</sup>にしたがうとする。VSLMにおいて、locality\*\* 外のページが参照される確率をλ, locality sizeをu, プログラム・サイズをv ( $u \leq v$ ) とすると、locality 内のあるページが参照される確率は  $(1-\lambda)/u$ , locality 外のあるページが参照される確率は  $\lambda/(v-u)$  である。locality 外のページが参照された場合、当ページは新たな locality に含まれ、古い locality の中から1ページがランダムに選ばれて locality から除外される。

working-set scheduling のもとで適当に window-size を選ぶと、working-set が locality の良い近似となることが知られている<sup>3)</sup>。したがって以下、両者が一致すると仮定し、working-set size w がuと等しいとする。

会話型ジョブの1 interaction のページ参照回数をdとすれば、この間のpage-fault 発生回数の期待値は  $d\lambda$  である。ここで、

$$r \triangleq d\lambda/w, \quad (17)$$

とおくと、あるページが1 interaction を通じて working-set 内にとどまる確率は、rが比較的小さいとき、次式で近似できる。

$$\begin{aligned} (1-\lambda/w)^d &= 1 - d\lambda/w + (1/2)(\lambda/w)^2 d(d-1) - \dots \\ &\cong 1 - r. \end{aligned} \quad (18)$$

誤差は  $r^2/2$  のオーダーである。なお、いったん working-set から外れたページは、その interaction 中に再び working-set に加わることはないとする。

以上の準備のもとにxを求める。working-set は、interaction 開始時に swap-in され、終了時に swap-out される\*\*\*。xは swap-in cost に関与するから、interaction 開始直前、いいかえれば swap-out 完了後の値を求めればよい。

(1) working-set 内のページのスロット位置 (そ

の1)

一般に working-set 内のページは、それが working-set に加わって以来少なくとも一回書きこみが行われた(チェンジした\*)か否か、またそのスロットが page-out により割り当てられたか swap-out により割り当てられたかにより、4つに分類できる。ただし、page-out は working-set から外れたとき行われるので、working-set に加わってからチェンジし、なお working-set 内にとどまっているページは、すべて swap-out によりスロットを割り当てられたものである。したがって実際には3つに分類できる。まず、チェンジし、かつ swap-out によりスロットを割り当てられたページ群につき考える。

interaction 開始直前において、前回の interaction 中にチェンジした  $w p$  ページのスロット群は1ブロックを形成している。一般に第  $(i+1)$  回前の interaction 中にチェンジし、その後チェンジせず working-set にとどまったページのスロット群はブロックを形成する。その大きさ\*\*を  $E_i$  とすると、

$$E_i = w p (1-p)^i (1-r)^i \quad (i=0, 1, 2, \dots) \quad (19)$$

となり、和は次式で与えられる。

$$\sum_{i=0}^{\infty} E_i = w p / (p+r-p r). \quad (20)$$

ここで次のようにしてブロック数を求める。 $E_i \geq 1$  なら、当ブロックの存在する確率を1と考える。 $E_i < 1$  なら、当ブロックの存在する確率を  $E_i$  に等しいと考える(この近似については付録1で考察を加えた)。 $E_i$  は  $i$  に関し単調減少であるから、 $E_i \geq 1$  のなりたつ最大の  $i$  を  $I$  とかくと、ブロック数  $\Gamma$  は次式で与えられる。

$$\Gamma = I + \sum_{i=I+1}^{\infty} E_i. \quad (21)$$

次に、working-set に加わって以来、一度もチェンジしていないページ群につき考える。このページ群は、そのスロットが page-out で割り当てられたものと、swap-out で割り当てられたものからなる。

前回の interaction 中に working-set に加わり、チェンジしなかったページ数は  $w r (1-p)$  である。一般に、第  $(i+1)$  回前の interaction 中に working-set に加わり、その後チェンジせずに working-set にとどまったページ数を  $H_i$  とすると、

$$H_i = w r (1-p)^{i+1} (1-r)^i \quad (i=0, 1, 2, \dots) \quad (22)$$

となり、和は次式で与えられる。

$$\sum_{i=0}^{\infty} H_i = w r (1-p) / (p+r-p r). \quad (23)$$

式(20)と式(23)を加えると、その和は  $w$  に一致する。

$E_i$  と異なり、 $H_i$  に対応するスロット群は必ずしも同一ブロックに属するわけではない。 $H_i$  に対応するスロット群の位置を解析するためには、working-set 外のページのスロット位置に着目する必要がある。

(2) working-set 外のページのスロット位置

interaction 開始時に working-set 外にあったあるページが、interaction 終了時まで引き続き working-set 外にとどまる確率は、

$$\{1 - \lambda / (v-w)\}^d \cong 1 - w r / (v-w) \quad (24)$$

である(誤差は  $r^2/2$  のオーダー)。いま、

$$q \triangleq w r / (v-w) \quad (25)$$

とすれば、 $q$  は working-set 外のページが1 interaction の間に working-set に加わる確率である。なお、いったん working-set に加わったページは、その interaction 中は必ず working-set 内にとどまると仮定する。

working-set 外のページは、そのスロットが page-out により割り当てられたか swap-out により割り当てられたかにより、2つに分類できる。まず前者について考える。

前回の interaction で page-out されたページ群の形成するブロックの大きさ  $\zeta_0$  は、

$$\zeta_0 = w p r \quad (26)$$

である。次回、および次々回の interaction 終了時の、当該ブロックの大きさをそれぞれ  $\zeta_1, \zeta_2$  とすると、それらは次のように与えられる。

$$\zeta_1 = \zeta_0 (1-q) \quad (27)$$

$$\zeta_2 = \zeta_1 (1-q) + \zeta_0 q (1-p)^2 r. \quad (28)$$

式(28)の第2項は、working-set に入っていた  $\zeta_0 q$  ページの中の  $\zeta_0 q (1-p)^2 r$  ページは、チェンジせずに再び working-set から外れたことを表す。一般に、 $n$  回 interaction 終了時の当該ブロックの大きさ  $\zeta_n$  は、

$$\zeta_n = \zeta_{n-1} (1-q) + \sum_{i=0}^{n-2} q (1-p)^{n-i} \times (1-r)^{n-i-2} \cdot r \cdot \zeta_i \quad (n=2, 3, \dots) \quad (29)$$

で与えられる。 $\zeta_n$  は、いいかえると、 $(n+1)$  回前の interaction で page-out されたページのうち、現在 working-set に含まれないもののスロット群からなるブロックの大きさである。なお、ここで次式が成り立

\*  $p=0$  の場合の  $\lambda$  は明らかなので、ここで  $p>0$  と仮定する。

\*\* 以下、「ブロックの大きさ」はすべて期待値を表す。

つ (証明は付録2).

$$\sum_{n=0}^{\infty} \zeta_n = (v-w)(p+r-pr)/(1+r-pr). \quad (30)$$

次に後者について考える. いま, 前々回の interaction 終了時に swap-out され, 前回の interaction ではチェンジせずに working-set から外れたページのスロット群の形成するブロックの大きさ  $\eta_0$  は,

$$\eta_0 = wp(1-p)r \quad (31)$$

である. 次回, および次々回の interaction 終了時の当該ブロックの大きさを  $\eta_1, \eta_2$  とすると,

$$\eta_1 = \eta_0(1-q) + wp(1-p)^2(1-r)r, \quad (32)$$

$$\eta_2 = \eta_1(1-q) + \eta_0q(1-p)^2r + wp(1-p)^3(1-r)^2r \quad (33)$$

となる. 式(32)の第2項, 式(33)の第3項は, 前々回の interaction 終了時に swap-out され, チェンジせずに working-set 内にとどまっていたが, 各々次回ないし次々回の interaction で working-set から外れたページに対応する. 一般に,  $n$ 回 interaction 終了時の当該ブロックの大きさを  $\eta_n$  とすれば, 次式が成り立つ.

$$\eta_n = \eta_{n-1}(1-q) + \sum_{i=0}^{n-2} q(1-p)^{n-i} \cdot (1-r)^{n-i-2} r \eta_i + wp(1-p)^{n+1} \cdot (1-r)^n r \quad (n=2, 3, \dots) \quad (34)$$

$\eta_n$  は, いいかえると,  $(n+2)$  回前の interaction 終了時に swap-out され,  $(n+1)$  回前以降の interaction で, チェンジせずに working-set から外れたページのスロット群からなるブロックの大きさである. なお, ここで次式が成り立つ (証明は付録2).

$$\sum_{n=0}^{\infty} \eta_n = (v-w)(1-p)/(1+r-pr). \quad (35)$$

式(30)と式(35)の和は  $(v-w)$  に一致する.

以上より, 定常状態で, working-set 外のページのスロットは, 大きさ  $\zeta_n$  ないし  $\eta_n$  ( $n=0, 1, 2, \dots$ ) の相異なるブロックのいずれかに属していることがわかる.

(3) working-set 内のページのスロット位置 (その2)

再び  $\sum_{i=0}^{\infty} H_i$  ページにつき考える. 既述のように, これはそのスロット割り当ての契機が page-out によるものと, swap-out によるものとに分類できる. まず, page-out でスロットを割り当てられたページ群に着目する.  $H_i$  は,  $(i+1)$  回前の interaction 開始時に working-set 外にあったページの中からランダム

に選ばれたページ群の大きさである. したがって, 大きさ  $\zeta_n$  のページ群の中で  $H_i$  に含まれるページ数は  $H_i \cdot \zeta_n / (v-w)$  である. ここで, ある時点における大きさ  $\zeta_n$  のブロックは, その前回の interaction 開始直前には大きさ  $\zeta_{n-1}$  のブロックであったわけだから,  $H_i \cdot \zeta_n / (v-w)$  ページと  $H_{i+1} \cdot \zeta_{n-1} / (v-w)$  ページとは, そのスロットが同一のブロックに属する. 一般に, 大きさ  $H_{m-j-1} \cdot \zeta_j / (v-w)$  ( $j=0, 1, 2, \dots, m-1$ ) のスロット群は同一のブロックに属する. 結局ブロックの大きさ  $\phi_m$  は次式で与えられる.

$$\begin{aligned} \phi_m &= \sum_{j=0}^{m-1} H_{m-j-1} \cdot \zeta_j / (v-w) \\ &= wr(1-p)(v-w)^{-1} \sum_{j=0}^{m-1} (1-p)^{m-j-1} \\ &\quad \times (1-r)^{m-j-1} \cdot \zeta_j \quad (m=1, 2, \dots) \quad (36) \end{aligned}$$

なお  $\phi_m$  の和は次式で与えられる (証明は付録3).

$$\sum_{m=1}^{\infty} \phi_m = wr(1-p)/(1+r-pr). \quad (37)$$

$\phi_m \geq 1$  の成り立つ  $m$  の添字集合を  $\{M\}$  とすると, スロットの属する相異なるブロック数  $\Pi$  は,

$$\Pi = \sum_{m \in \{M\}} 1 + \sum_{m \in \{M\}} \phi_m \quad (38)$$

で与えられる (付録1参照).  $\{M\}$  を求めるには,

$$\sum_{m=1}^{\bar{m}} \phi_m > \sum_{m=1}^{\infty} \phi_m - 1 \quad (39)$$

が成立する最小の  $\bar{m}$  を  $\hat{m}$  とかくとき,  $\phi_1 \rightarrow \phi_2 \rightarrow \dots \rightarrow \phi_{\hat{m}}$  を順次計算し, その中で  $\phi_m \geq 1$  なるものを選べばよい. 更に次式が成り立つので, 式(38)が計算できる.

$$\sum_{m \in \{M\}} \phi_m = \sum_{m=1}^{\infty} \phi_m - \sum_{m \in \{M\}} \phi_m. \quad (40)$$

なお,  $\phi_m$  は次のように表わせるので,  $\zeta_n$  を求めることなく  $\phi_1 \rightarrow \phi_2 \rightarrow \phi_3 \rightarrow \dots$  と順次計算することができる (証明は付録4).

$$\phi_1 = w^2 r^2 p(1-p)/(v-w). \quad (41)$$

$$\begin{aligned} \phi_2 &= w^2 r^2 p(1-p) \{ (1-p)(1-r) + (1-q) \} \\ &\quad \div (v-w). \quad (42) \end{aligned}$$

$$\begin{aligned} \phi_m &= qr \sum_{j=1}^{m-2} (1-p)^{m-j} \cdot (1-r)^{m-i-2} \cdot \phi_j \\ &\quad + (1-q)\phi_{m-1} + w^2 r^2 p(1-p)^m (1-r)^{m-1} \\ &\quad \div (v-w) \quad (m=3, 4, \dots). \quad (43) \end{aligned}$$

次に,  $\sum_{i=0}^{\infty} H_i$  ページの中で, swap-out でスロットを割り当てられたページ群に着目する. このとき,  $\zeta_n$  を  $\eta_n$  とかけば全く同様な議論が成り立ち, ブロックの大きさ  $\phi_m$  は,

$$\begin{aligned} \phi_m &= \sum_{j=0}^{m-1} H_{m-j-1} \cdot \eta_j / (v-w) \\ &= wr(1-p)(v-w)^{-1} \sum_{j=0}^{m-1} (1-p)^{m-j-1} \\ &\quad \times (1-r)^{m-j-1} \cdot \eta_j \quad (m=1, 2, \dots) \end{aligned} \quad (44)$$

となり、その和は次式で与えられる(証明は付録4)。

$$\sum_{m=1}^{\infty} \phi_m = wr(1-p)^2 / \{(p+r-pr)(1+r-pr)\} \quad (45)$$

式(37)と式(45)の和は、確かに  $\sum_{i=0}^{\infty} H_i$  に一致する。

$\phi_m \geq 1$  の成り立つ  $m$  の添字集合を  $\{M'\}$  とすると、スロットの属する相異なるブロック数  $\Pi'$  は、

$$\Pi' = \sum_{m \in \{M'\}} 1 + \sum_{m \notin \{M'\}} \phi_m \quad (46)$$

で与えられる。式(46)は式(38)と全く同様にして計算できる。なお、 $\phi_m$  は次のように表わせるので、 $\eta_m$  を求めることなく  $\phi_1 \rightarrow \phi_2 \rightarrow \phi_3 \rightarrow \dots$  と順次計算することができる(証明は付録4)。

$$\phi_1 = w^2 r^2 p(1-p)^2 / (v-w) \quad (47)$$

$$\begin{aligned} \phi_2 &= w^2 r^2 p(1-p)^2 \{2(1-p)(1-r) + (1-q)\} \\ &\quad \div (v-w) \end{aligned} \quad (48)$$

$$\begin{aligned} \phi_m &= qr \sum_{j=1}^{m-2} (1-p)^{m-j} \cdot (1-r)^{m-j-2} \cdot \phi_j \\ &\quad + (1-q)\phi_{m-1} + mw^2 r^2 p(1-p)^{m+1} \cdot (1-r)^{m-1} \\ &\quad \div (v-w) \quad (m=3, 4, \dots) \end{aligned} \quad (49)$$

(4)  $x$  の算出

(1)~(3)の議論から、結局 working-set 内のページ群に割り当てられたスロット群の属する相異なるブロック数の期待値  $x$  は、式(21), (38), (46)より次のように算出される。

$$x = \Gamma + \Pi + \Pi' \quad (50)$$

ただし前述のように、 $\Gamma$  は working-set に加わって以来少なくとも一回チェンジしたページ群、 $(\Pi + \Pi')$  は全くチェンジしなかったページ群に対応する。さらに  $(\Pi + \Pi')$  の中で、 $\Pi$  は page-out によりスロットを割り当てられたページ群、 $\Pi'$  は swap-out によりスロットを割り当てられたページ群に対応する。

5. 数値例

パラメータの値を次のように定めて、固定方式と浮動方式の性能比較を行った。  $w=20$  ページ、 $v=40$  ページ、 $p=0.2, 0.4, 0.6, 0.8$ 、 $r=0.1, 0.2$ 、 $L=100$

\* H-8589-1 ディスタを想定する。  
\*\* 式(50)で、 $v$  は  $x$  が整数値のときのみ計算できる。したがって、表1の  $v$  は内挿した値である。

表1 二次記憶(ディスク)上における、ワーキングセットに割り当てられたスロット群の散在度  
Table 1 The dispersion of slots over a paging device.

| $p$ | $r$ | $x$  | $v$  | $x' \pm \sigma'$ |
|-----|-----|------|------|------------------|
| 0.2 | 0.1 | 13.5 | 12.7 | 11.7±2.3         |
| 0.2 | 0.2 | 14.8 | 13.8 | 12.5±2.2         |
| 0.4 | 0.1 | 8.1  | 7.8  | 7.3±2.0          |
| 0.4 | 0.2 | 9.3  | 8.9  | 8.2±1.8          |
| 0.6 | 0.1 | 5.1  | 5.0  | 5.0±1.5          |
| 0.6 | 0.2 | 5.9  | 5.8  | 5.6±1.4          |
| 0.8 | 0.1 | 3.1  | 3.1  | 3.1±1.0          |
| 0.8 | 0.2 | 3.4  | 3.4  | 3.3±1.1          |

シリンドラ、 $S_0=10 \text{ ms}^{7)}$ 、 $K=0.1125 \text{ ms}^{7)}/\text{シリンドラ}$ 、 $R=16.7 \text{ ms}$ 。

式(50)により求めた  $x$  と、この  $x$  と式(13)より求めた  $y$  とを表1に示す。さらに表1には、解析結果を検証するために実行したシミュレーションより求めた、ブロック数の期待値  $x'$  とその標準偏差  $\sigma'$  も示した。

なお、シミュレーションの概略は次の通りである。  
 $v$  ページの各々には、working-set 内か外かを示すエントリと、その割り当てられたスロットの属するブロック番号が与えられる。初期状態では、全ページのブロック番号は0であり、ランダムに選ばれた  $w$  ページが working-set 内、残りの  $(v-w)$  ページが working-set 外とする。シミュレーションは interaction を繰り返すことにより行われるが、ここで interaction は、swap-in→page-in→page-out→swap-out の4段階から構成される。swap-in では、各ページのエントリは変化しない。page-in では、working-set 外の各ページを  $q$  の確率で working-set 内に含める。次に working-set 内の各ページを  $p$  の確率でチェンジする。page-out では、新しく page-in されたページ以外の working-set 内の各ページを  $r$  の確率で working-set 外に出す。このとき出されるページ群の中でチェンジしたページ群に新たなブロック番号を与える。swap-out では、working-set 内のページ群の中でチェンジしたページ群に新たなブロック番号を与える。

interaction を多数回くり返した後、working-set 内のページ群に対応する相異なるブロック番号の数を算定した。  $x'$  と  $\sigma'$  は、100回以上 interaction をくり返して定常状態になった時点でのブロック数の、100サンプルについて求めた。  $x$  はほぼ  $x' \pm \sigma'$  の範囲に含まれており、  $x$  と  $x'$  との差異は 0~16% である。  $x$  は  $x'$  よりやや大きい、これは付録1で考察した理由によると考えられる。

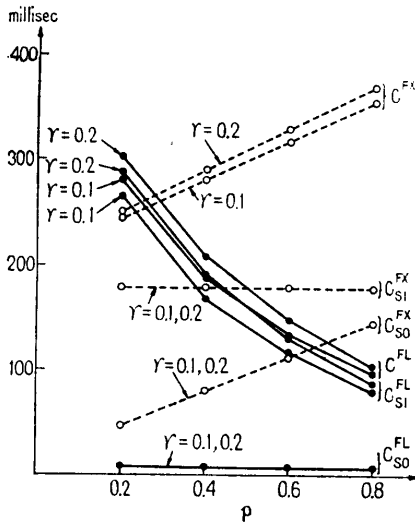


図3 スワップ・アウト・コスト  $C_{so}$ , スワップ・イン・コスト  $C_{si}$ , トータル・コスト  $C$  の各々に関する, 固定方式 (○) と浮動方式 (●) との性能比較

Fig. 3 Performance comparison of the two techniques (Fixed: (○), Floating: (●)) with respect to the swap-out cost  $C_{so}$ , swap-in cost  $C_{si}$ , total cost  $C$ .

両方式の swap-out cost, swap-in cost, total cost を比較した結果を, 図3に示す. swap-out cost は浮動方式がすぐれている. swap-in cost は, チェンジする確率  $\rho$  が小さいとき固定方式がすぐれている. さらに total cost を比べると, 総合的な効率としては,  $\rho$  が小さいとき固定方式, 大きいとき浮動方式がすぐれていることがわかる. これは,  $\rho$  が増大するにつれ  $x$  や  $y$  が減少し, 浮動方式が有利になるためである.

## 6. むすび

仮想メモリの二次記憶の割り当てに関して, 代表的な方式である固定方式と浮動方式につき, その性能を比較するための解析モデルを示した. 特に, 浮動方式のもとでは, ひとつのプログラムの working-set 内のページのスロット群が二次記憶媒体上の各所に散らばるが, この散らばりの度合を算定した. 算定結果をシミュレーションにより検証したところ, 差異は 0~16%であった.

具体的数値例により, 両方式の効率を比較したところ, 上記の散らばりが小さい場合は浮動方式, 大きい場合は固定方式が, それぞれすぐれていた.

今後は, 実測による検証も含め, 解析モデルの妥当性に関する検討を続ける予定である.

終りに, 本研究について御指導いただいた東京大学大須賀節雄助教授, 筑波大学益田隆司助教授に深く感謝致します.

## 参考文献

- 1) Denning, P. J.: Effects of scheduling on file memory operations, Proc. SJCC, pp. 9-21 (1967).
- 2) Teorey, T. J., Pinkerton, T. B.: A Comparative Analysis of Disk Scheduling Policies, Commun. ACM, Vol. 15, No. 3, pp. 177-184 (1972).
- 3) Spirn, J. R., Denning, P. J.: Experiments with program locality, Proc. FJCC, pp. 611-621 (1972).
- 4) Denning, P. J., Graham, G. S.: Multiprogrammed Memory Management, Proc. IEEE, Vol. 63, No. 6, pp. 924-939 (1975).
- 5) Denning, P. J.: The Working Set Model for Program Behavior, Commun. ACM, Vol. 5, No. 11, pp. 323-333 (1968).
- 6) W. フェラー: 確率論とその応用(上), 河田龍夫監訳, 紀伊国屋書店 (1969).
- 7) 日立製作所: H-8549-2 ディスク制御装置, H-8589 ディスク駆動装置, ハードウェア・マニュアル, 8080-2-007 (1974).
- 8) 日立製作所: VOS2システムプログラムの手引, プログラム・マニュアル, 8080-3-002 (1975).
- 9) 日立製作所: VOS3システムプログラムの手引, プログラム・マニュアル, 8090-3-002 (1976).
- 10) IBM: Introduction to Virtual Storage in System/370, Student Text (1972).

## 付 録

(1) 付録 1: ブロック数算定の近似について  
第  $i$  ブロックに  $n$  ページ存在する確率を  $z_i(n)$  とする ( $n=0, 1, 2, \dots$ ). 当ブロックが存在するためには, 1 ページ以上あればよいから, 真に求める値  $\alpha_i$  は,

$$\alpha_i \triangleq \sum_{n=1}^{\infty} z_i(n) = 1 - z_i(0) \quad (i)$$

で与えられる. しかし, 一般に  $z_i(n)$  は不明であり, 与えられているのは, ブロック・サイズの期待値  $E_i$  すなわち  $\sum_{n=0}^{\infty} n \cdot z_i(n)$  のみである. ここで,

- $E_i \geq 1$  のとき,  $z_i(0) = 0$  と仮定すると  $\alpha_i = 1$ ,
- $E_i < 1$  のとき,  $z_i(n) = 0$  ( $n \geq 2$ ) と仮定すると  $\alpha_i = E_i$ .

となる. なお, 一般には  $z_i(n) \geq 0$  ( $n \geq 0$ ) なので, 上記の近似は  $\alpha_i$  をやや過大評価することになる.(この議論は,  $E_i$  のみならず,  $\phi_m, \psi_m$  についても同様.)

(2) 付録 2: 式(30), (35)の証明

$\zeta_n$  の定義より各項は非負でありその和  $\sum_{n=0}^{\infty} \zeta_n$  が 0 以上で  $v$  をこえないことは明らかである. 式(26)~





$$\begin{aligned} \phi_m = & q r \sum_{j'=1}^{m-2} \{(1-p)^{m-j'} \cdot (1-r)^{m-j'-2} \cdot \phi_{j'}\} \\ & + (1-q) \phi_{m-1} + w r (1-p)^m (1-r)^{m-1} \\ & \times (v-w)^{-1} \zeta_0 \end{aligned} \quad (\text{xii})$$

となる。式(xii)に式(26)を用いると式(43)が得られる。

次に、式(47)~(49)については、式(31), (32)を式(44)に用いれば式(47), (48)はただちに得られる。さらに、 $\phi_m$  ( $m \geq 3$ )は、式(34)を式(44)に用いることにより、次のようにかける。

$$\begin{aligned} \phi_m = & w r (1-p)(v-w)^{-1} \\ & \times \sum_{j=2}^{m-1} \left[ (1-p)^{m-j-1} \cdot (1-r)^{m-j-1} \right. \\ & \times \left. \left\{ \sum_{i=0}^{j-2} q (1-p)^{j-i} \cdot (1-r)^{j-i-2} \cdot r \cdot \eta_i \right. \right. \\ & \left. \left. + \eta_{j-1} (1-q) + \eta_0 (1-p)^j (1-r)^j \right\} \right] \\ & + w r (1-p)(v-w)^{-1} \\ & \times \{(1-p)^{m-2} (1-r)^{m-2} \eta_1 \\ & + (1-p)^{m-1} (1-r)^{m-1} \eta_0\} \end{aligned} \quad (\text{xiii})$$

式(xiii)の第1項に着目し、これを $\Delta\phi$ とおく。上記議論で、 $\zeta$ を $\eta$ でおきかえ、式(32), (44)を用いると、次式が得られる。

$$\begin{aligned} \Delta\phi = & q r \sum_{j'=1}^{m-2} \{(1-p)^{m-j'} \cdot (1-r)^{m-j'-2} \cdot \phi_{j'}\} \\ & + (1-q) \phi_{m-1} - w r (1-p)(v-w)^{-1} \\ & \times (1-p)^{m-2} (1-r)^{m-2} (1-q) \eta_0 \\ & + w r (1-p)(v-w)^{-1} \\ & \times \sum_{j'=1}^{m-2} \{(1-p)^{m-1} (1-r)^{m-1} \eta_0\} \\ = & q r \sum_{j'=1}^{m-2} \{(1-p)^{m-j'} \cdot (1-r)^{m-j'-2} \cdot \phi_{j'}\} \\ & + (1-q) \phi_{m-1} - \{w r (v-w)^{-1} (1-p)^{m-1} \\ & \times (1-r)^{m-2} \eta_1 - w^2 r^2 p (1-p)^{m+1} \\ & \times (1-r)^{m-1} (v-w)^{-1}\} + (m-2) w r (1-p)^m \\ & \times (1-r)^{m-1} (v-w)^{-1} \eta_0. \end{aligned} \quad (\text{xvi})$$

式(xvi)を式(xiii)に代入し、式(31)を用いれば、ただちに式(49)が得られる。

(昭和53年5月16日受付)

(昭和54年1月18日採録)